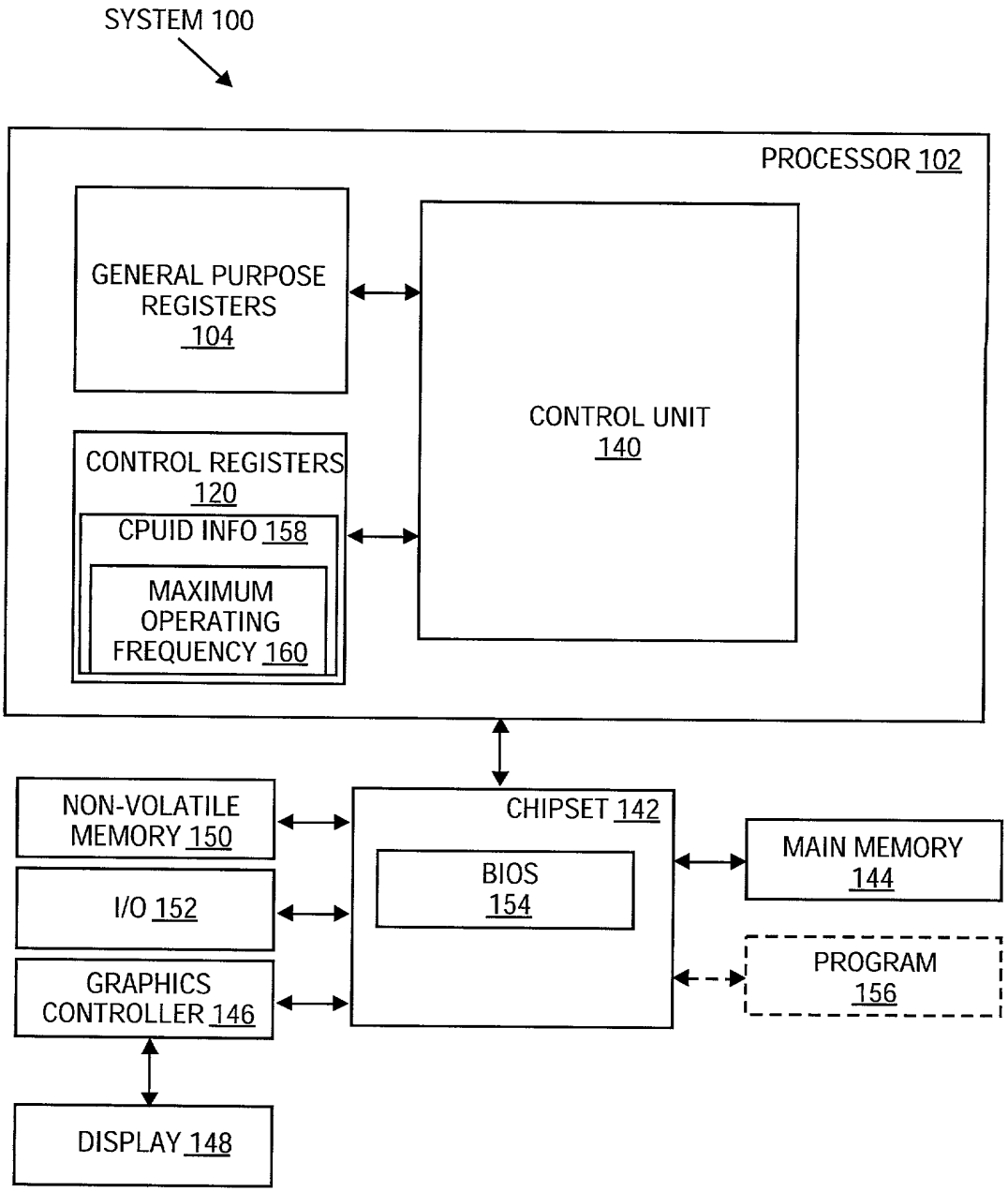

SYSTEM 100
PROCESSOR 102
GENERAL PURPOSE REGISTERS 104
CONTROL UNIT 140
CONTROL REGISTERS 120
CPUID INFO 158
MAXIMUM OPERATING FREQUENCY 160
NON-VOLATILE MEMORY 150
I/O 152
GRAPHICS CONTROLLER 146
DISPLAY 148
CHIPSET 142
BIOS 154
MAIN MEMORY 144
PROGRAM 156

FIG. 1

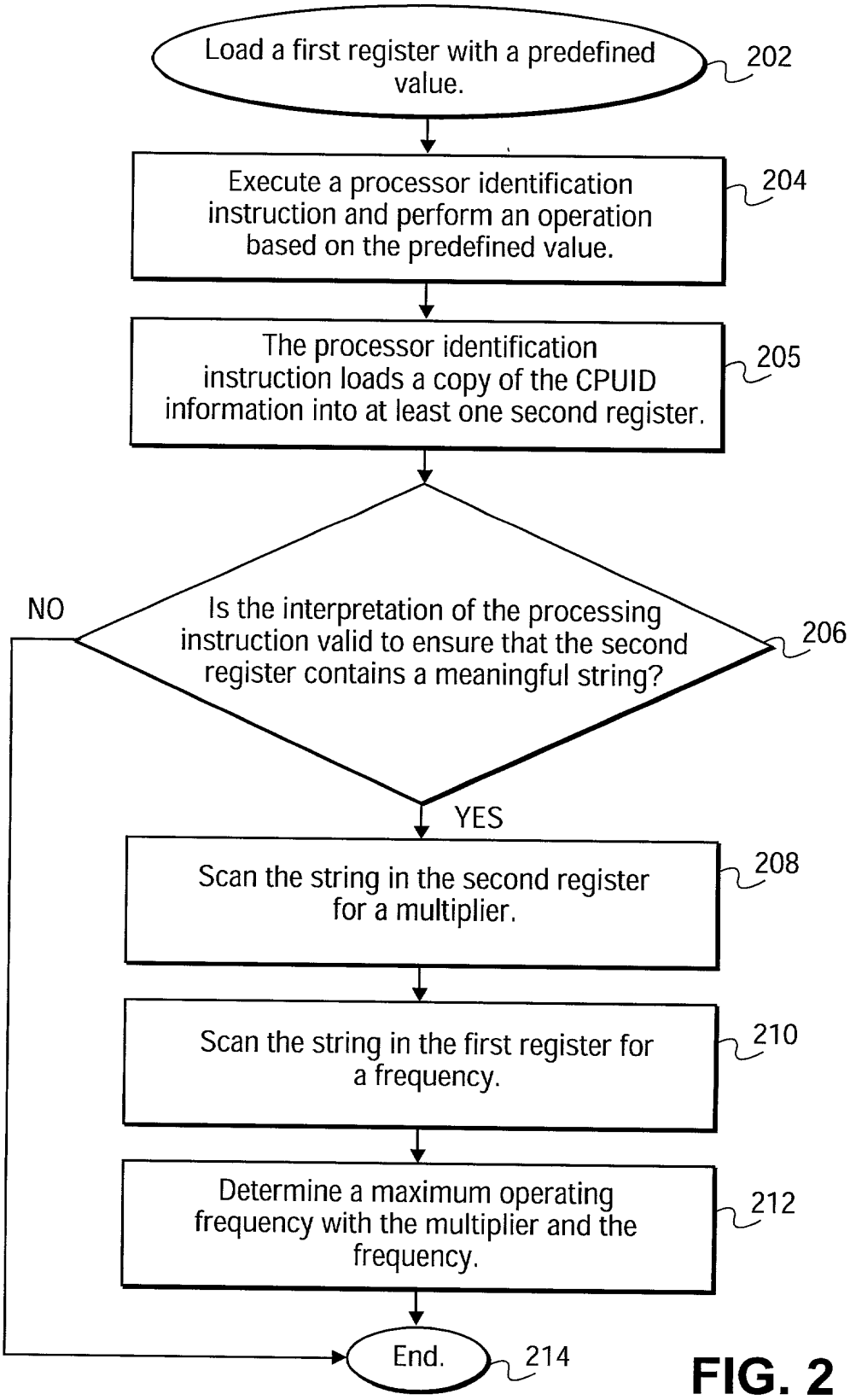

Load a first register with a predefined value.
202
Execute a processor identification instruction and perform an operation based on the predefined value.
204
The processor identification instruction loads a copy of the CPUID information into at least one second register.
205
Is the interpretation of the processing instruction valid to ensure that the second register contains a meaningful string?
206
NO
YES
Scan the string in the second register for a multiplier.
208
Scan the string in the first register for a frequency.
210
Determine a maximum operating frequency with the multiplier and the frequency.
212
End.
214

FIG. 2

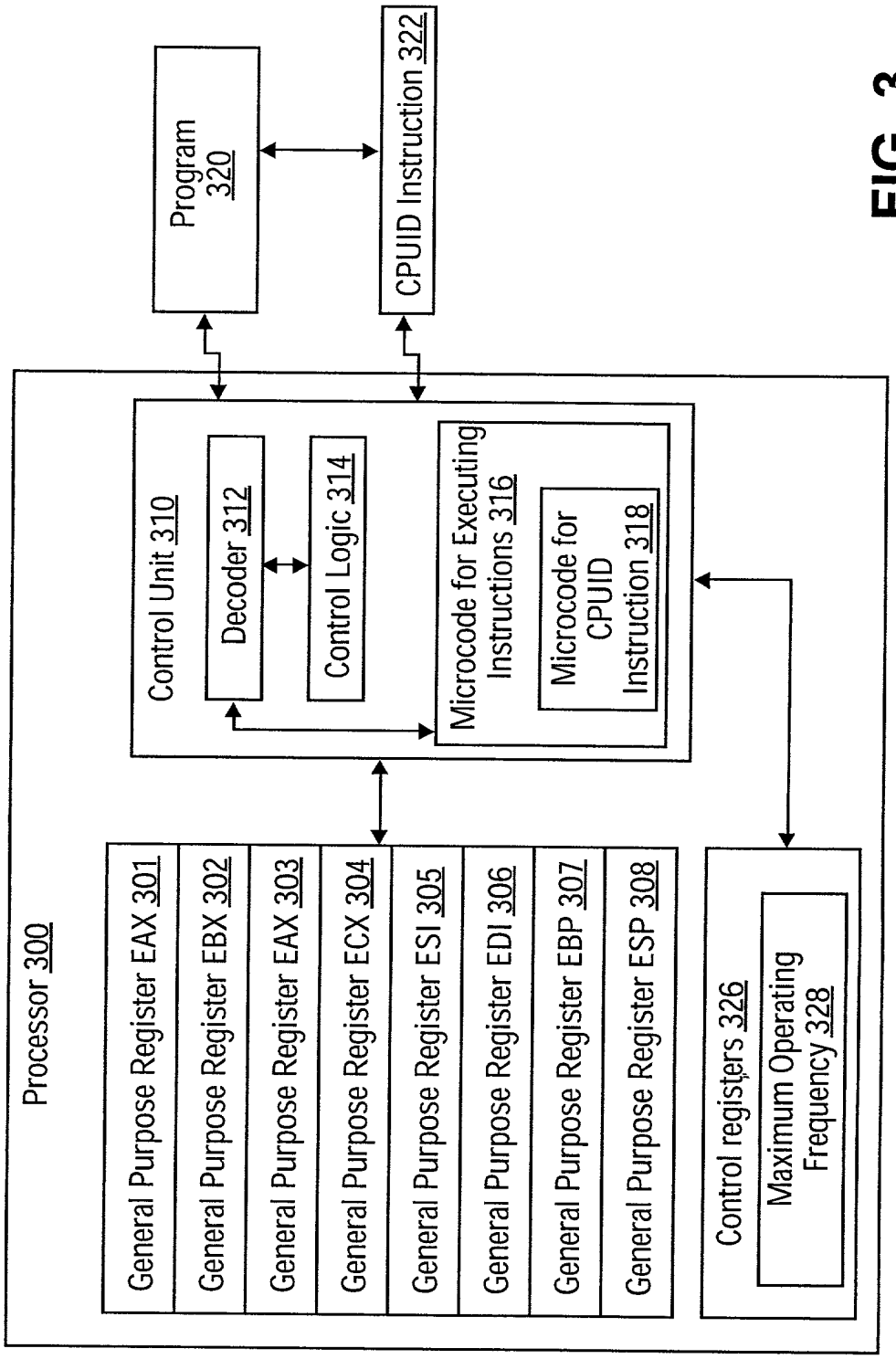

Program 320
CPUID Instruction 322
Processor 300
Control Unit 310
Decoder 312
Control Logic 314
Microcode for Executing Instructions 316
Microcode for CPUID Instruction 318
General Purpose Register EAX 301
General Purpose Register EBX 302
General Purpose Register EAX 303
General Purpose Register ECX 304
General Purpose Register ESI 305
General Purpose Register EDI 306
General Purpose Register EBP 307
General Purpose Register ESP 308
Control registers 326
Maximum Operating Frequency 328

FIG. 3

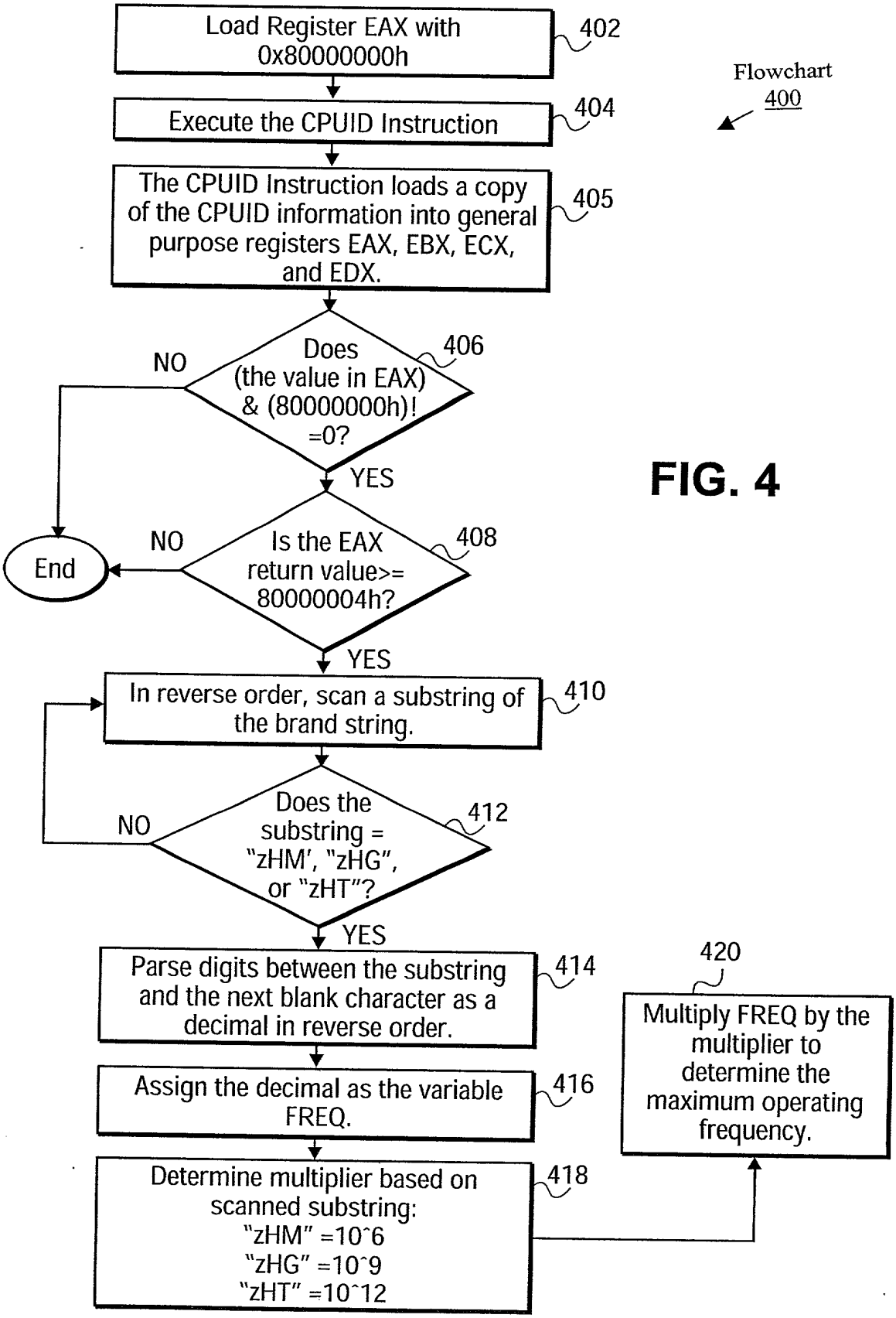
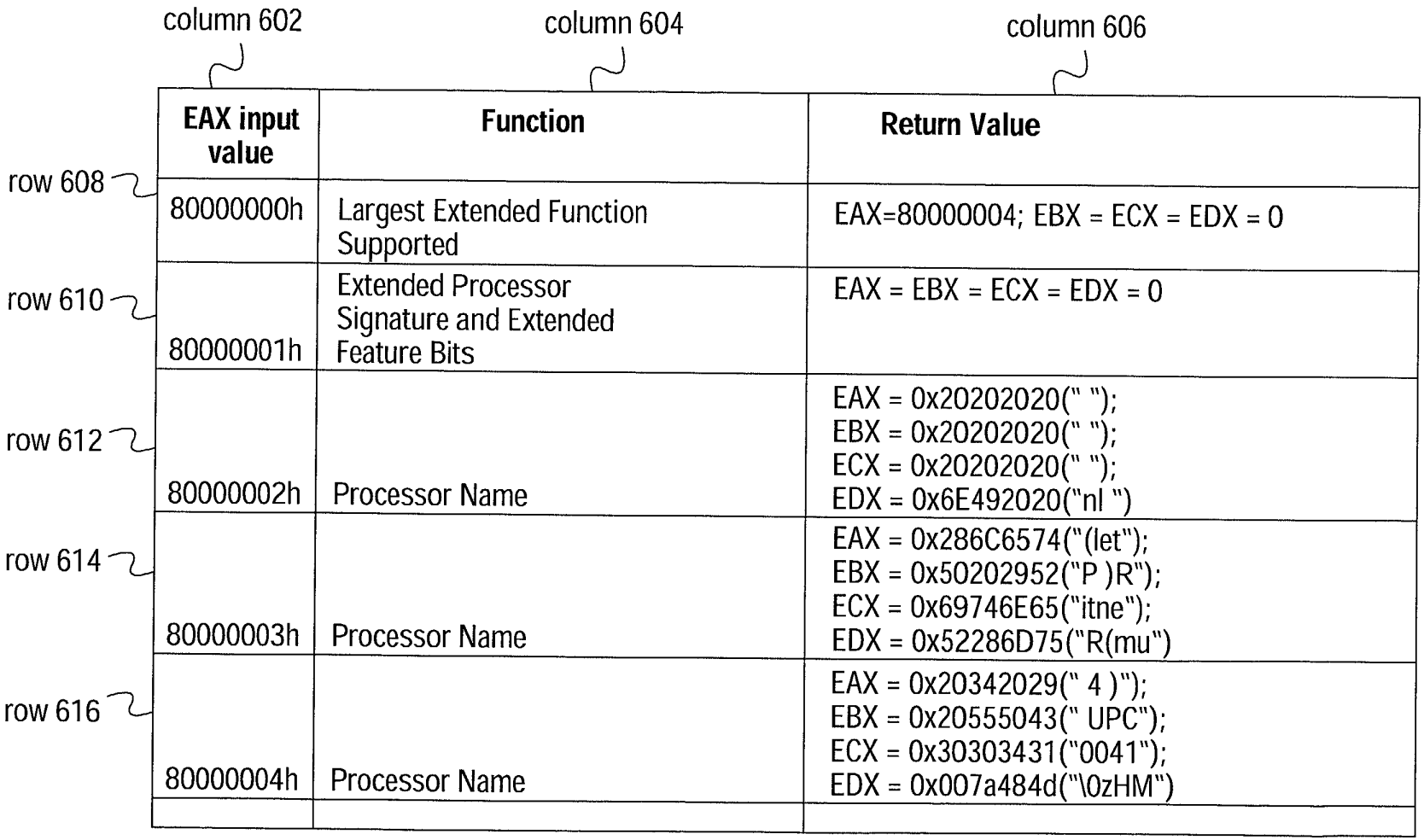

Load Register EAX with 0x80000000h
402
Execute the CPUID Instruction
404
The CPUID Instruction loads a copy of the CPUID information into general purpose registers EAX, EBX, ECX, and EDX.
405
Does (the value in EAX) & (80000000h)! =0?
406
NO
YES
Is the EAX return value>= 80000004h?
408
NO
End
YES
In reverse order, scan a substring of the brand string.
410
Does the substring = "zHM', "zHG", or "zHT"?
412
NO
YES
Parse digits between the substring and the next blank character as a decimal in reverse order.
414
Assign the decimal as the variable FREQ.
416
Determine multiplier based on scanned substring:
"zHM" =10^6
"zHG" =10^9
"zHT" =10^12
418
420
Multiply FREQ by the multiplier to determine the maximum operating frequency.
Flowchart
400

FIG. 4

Title: Apparatus and Method to Identify A Maximum Operating Frequency of a Processor
Inventor(s): Bryant E. Bigbee, Shivnandan Kaushik, Frank Binns
Atty. Docket No.: 42390P12481

	column 502	column 504	column 506
	EAX input value	**Function**	**Return Value**
row 508	80000000h	Largest Extended Function Supported	EAX=80000004; EBX = ECX = EDX = 0
row 510	80000001h	Extended Processor Signature and Extended Feature Bits	EAX = EBX = ECX = EDX = 0
row 512	80000002h	Processor Brand String	EAX, EBX, ECX, EDX, contain ASCII brand string
row 514	80000003h	Processor Brand String	EAX, EBX, ECX, EDX, contain ASCII brand string
row 516	80000004h	Processor Brand String	EAX, EBX, ECX, EDX, contain ASCII brand string

FIG. 5



EAX input value	Function	Return Value
80000000h	Largest Extended Function Supported	EAX=80000004; EBX = ECX = EDX = 0
80000001h	Extended Processor Signature and Extended Feature Bits	EAX = EBX = ECX = EDX = 0
80000002h	Processor Name	EAX = 0x20202020(" "); EBX = 0x20202020(" "); ECX = 0x20202020(" "); EDX = 0x6E492020("nl ")
80000003h	Processor Name	EAX = 0x286C6574("(let"); EBX = 0x50202952("P)R"); ECX = 0x69746E65("itne"); EDX = 0x52286D75("R(mu")
80000004h	Processor Name	EAX = 0x20342029(" 4)"); EBX = 0x20555043(" UPC"); ECX = 0x30303431("0041"); EDX = 0x007a484d("\0zHM")

FIG. 6